Spotify dataset Analysis

Gaayathri.M

2023-02-21

**Data Description:**

The data set is taken from data.world and it is about Spotify songs from 2010 to 2019 by year.

**Attribute Explanation:**

Title: Song’s title.

Artist: Song’s artist.

Top genre: The genre of the track.

Year: Song's year in the Billboard.

Bpm: Beats Per Minute - The tempo of the song.

Nrgy: Energy- The energy of a song - the higher the value, the more energetic Song.

Dnce: Danceability - The higher the value, the easier it is to dance to this song.

Db: Loudness dB- The higher the value, the louder the song.

Live: Liveness - The higher the value, the more likely the song is a live recording.

Val: Valence - The higher the value, the more positive mood for the song.

Dur: Length - The duration of the song.

Acous: Acousticness - The higher the value the more acoustic the song is.

Spch: Speechiness - The higher the value the more spoken word the song contains.

Pop: Popularity- The higher the value the more popular the song is.

**Assumption:**

* I assume that most of the genre are dance pop because most of the artist prefer dance pop.
* I assume that in 2014 dance pop hits more because dance pop becomes trend in 2014.
* I assume that dance pop has high (bpm) because dance pop have high beats.
* I assume that energy level is also high in dance pop because when it comes to dance vibe level is high.
* I assume that dance pop have high valence because when we see dance videos it gives positive mood, positive vibe and many positive things.

**Hypothesis:**

* I am analyzing the genre using count function because histogram does not fit in genre (categorical data). I assume that most of the genre is dance pop its correct .
* Now I am going to analyze dance pop hits more in 2014 using histogram its correct.
* And then I assume that dance pop has high bpm its wrong dance pop have medium level of bpm.
* My assumption that the dance pop have high energy level its wrong it has almost high energy level but it does not have high energy level.
* I assume that dance pop have high valence its wrong it has medium valence.

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(readr)  
top10s <- read\_csv("top10s.csv")

## New names  
## • `` -> `...1`

## Rows: 603 Columns: 15  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (3): title, artist, top.genre  
## dbl (12): ...1, year, bpm, nrgy, dnce, dB, live, val, dur, acous, spch, pop  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

summary(top10s)

## ...1 title artist top.genre   
## Min. : 1.0 Length:603 Length:603 Length:603   
## 1st Qu.:151.5 Class :character Class :character Class :character   
## Median :302.0 Mode :character Mode :character Mode :character   
## Mean :302.0   
## 3rd Qu.:452.5   
## Max. :603.0   
## year bpm nrgy dnce   
## Min. :2010 Min. : 0.0 Min. : 0.0 Min. : 0.00   
## 1st Qu.:2013 1st Qu.:100.0 1st Qu.:61.0 1st Qu.:57.00   
## Median :2015 Median :120.0 Median :74.0 Median :66.00   
## Mean :2015 Mean :118.5 Mean :70.5 Mean :64.38   
## 3rd Qu.:2017 3rd Qu.:129.0 3rd Qu.:82.0 3rd Qu.:73.00   
## Max. :2019 Max. :206.0 Max. :98.0 Max. :97.00   
## dB live val dur   
## Min. :-60.000 Min. : 0.00 Min. : 0.00 Min. :134.0   
## 1st Qu.: -6.000 1st Qu.: 9.00 1st Qu.:35.00 1st Qu.:202.0   
## Median : -5.000 Median :12.00 Median :52.00 Median :221.0   
## Mean : -5.579 Mean :17.77 Mean :52.23 Mean :224.7   
## 3rd Qu.: -4.000 3rd Qu.:24.00 3rd Qu.:69.00 3rd Qu.:239.5   
## Max. : -2.000 Max. :74.00 Max. :98.00 Max. :424.0   
## acous spch pop   
## Min. : 0.00 Min. : 0.000 Min. : 0.00   
## 1st Qu.: 2.00 1st Qu.: 4.000 1st Qu.:60.00   
## Median : 6.00 Median : 5.000 Median :69.00   
## Mean :14.33 Mean : 8.358 Mean :66.52   
## 3rd Qu.:17.00 3rd Qu.: 9.000 3rd Qu.:76.00   
## Max. :99.00 Max. :48.000 Max. :99.00

str(top10s)

## spc\_tbl\_ [603 × 15] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ ...1 : num [1:603] 1 2 3 4 5 6 7 8 9 10 ...  
## $ title : chr [1:603] "Hey, Soul Sister" "Love The Way You Lie" "TiK ToK" "Bad Romance" ...  
## $ artist : chr [1:603] "Train" "Eminem" "Kesha" "Lady Gaga" ...  
## $ top.genre: chr [1:603] "neo mellow" "detroit hip hop" "dance pop" "dance pop" ...  
## $ year : num [1:603] 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...  
## $ bpm : num [1:603] 97 87 120 119 109 65 120 148 93 126 ...  
## $ nrgy : num [1:603] 89 93 84 92 84 86 78 76 37 72 ...  
## $ dnce : num [1:603] 67 75 76 70 64 73 75 52 48 79 ...  
## $ dB : num [1:603] -4 -5 -3 -4 -5 -5 -4 -6 -8 -4 ...  
## $ live : num [1:603] 8 52 29 8 9 11 4 12 12 7 ...  
## $ val : num [1:603] 80 64 71 71 43 54 82 38 14 61 ...  
## $ dur : num [1:603] 217 263 200 295 221 214 203 225 216 235 ...  
## $ acous : num [1:603] 19 24 10 0 2 4 0 7 74 13 ...  
## $ spch : num [1:603] 4 23 14 4 4 14 9 4 3 4 ...  
## $ pop : num [1:603] 83 82 80 79 78 77 77 77 76 73 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. ...1 = col\_double(),  
## .. title = col\_character(),  
## .. artist = col\_character(),  
## .. top.genre = col\_character(),  
## .. year = col\_double(),  
## .. bpm = col\_double(),  
## .. nrgy = col\_double(),  
## .. dnce = col\_double(),  
## .. dB = col\_double(),  
## .. live = col\_double(),  
## .. val = col\_double(),  
## .. dur = col\_double(),  
## .. acous = col\_double(),  
## .. spch = col\_double(),  
## .. pop = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

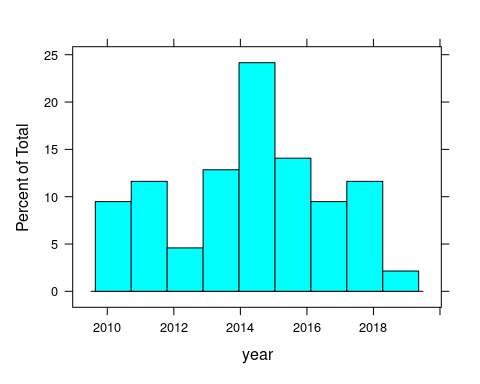
unique(top10s["artist"])

## # A tibble: 184 × 1  
## artist   
## <chr>   
## 1 Train   
## 2 Eminem   
## 3 Kesha   
## 4 Lady Gaga   
## 5 Bruno Mars   
## 6 Justin Bieber  
## 7 Taio Cruz   
## 8 OneRepublic   
## 9 Alicia Keys   
## 10 Rihanna   
## # … with 174 more rows

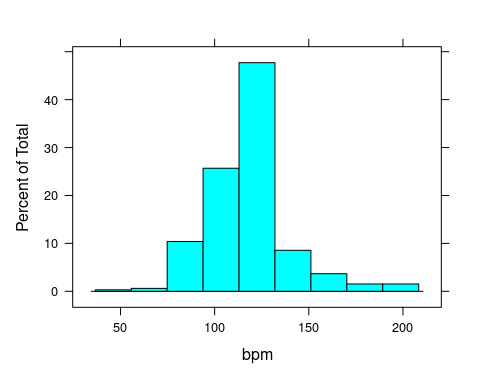
df=subset(top10s,top.genre=="dance pop")  
df

## # A tibble: 327 × 15  
## ...1 title artist top.g…¹ year bpm nrgy dnce dB live val dur  
## <dbl> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 3 TiK ToK Kesha dance … 2010 120 84 76 -3 29 71 200  
## 2 4 Bad Rom… Lady … dance … 2010 119 92 70 -4 8 71 295  
## 3 7 Dynamite Taio … dance … 2010 120 78 75 -4 4 82 203  
## 4 8 Secrets OneRe… dance … 2010 148 76 52 -6 12 38 225  
## 5 11 Club Ca… Flo R… dance … 2010 128 87 62 -4 6 47 235  
## 6 13 Cooler … Mike … dance … 2010 130 82 77 -5 70 63 213  
## 7 14 Telepho… Lady … dance … 2010 122 83 83 -6 11 71 221  
## 8 15 Like A … Far E… dance … 2010 125 84 44 -8 12 78 217  
## 9 17 Eenie M… Sean … dance … 2010 121 61 72 -4 11 83 202  
## 10 18 The Tim… The B… dance … 2010 128 81 82 -8 60 44 308  
## # … with 317 more rows, 3 more variables: acous <dbl>, spch <dbl>, pop <dbl>,  
## # and abbreviated variable name ¹​top.genre

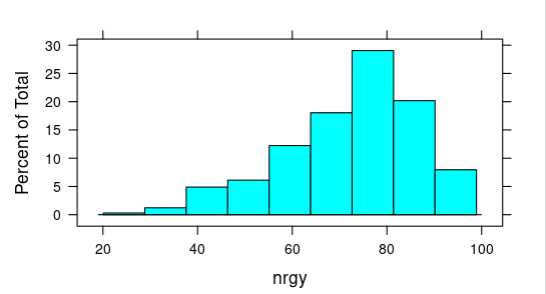
library(lattice)  
histogram(~year,df)



**Fig.1.1**

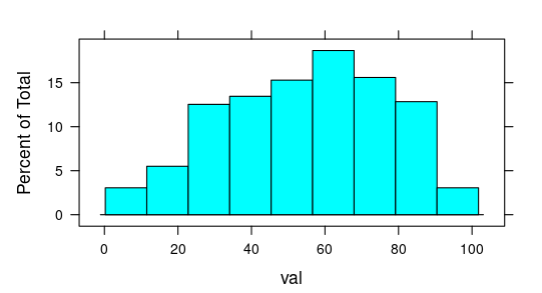
histogram(~bpm,df)

**Fig.1.2**

histogram(~nrgy,df) 

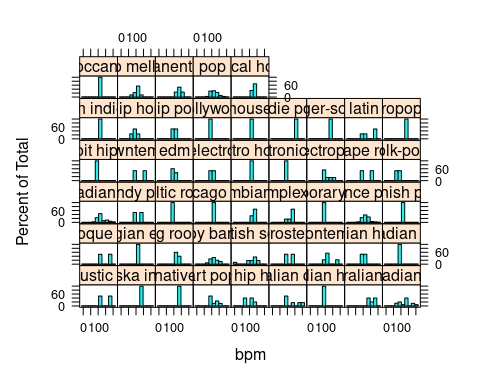
**Fig.1.3**

histogram(~val,df)



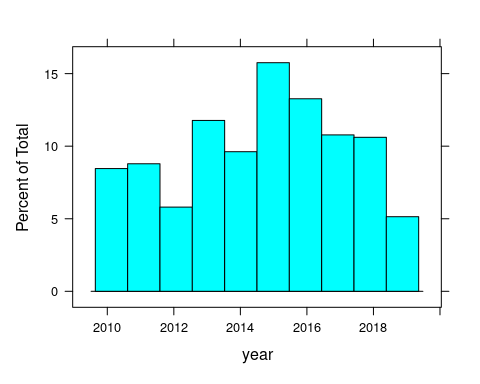
**Fig.1.4**

histogram(~bpm|top.genre,data=top10s)



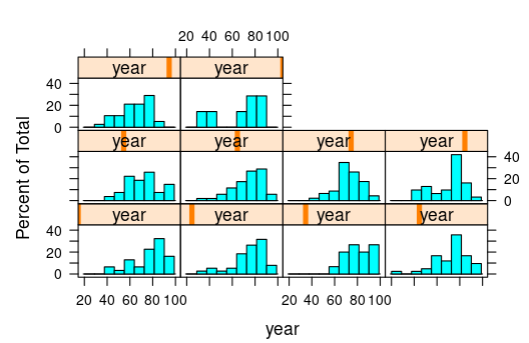
**Fig.1.5**

histogram(~year,data=top10s)



**Fig.1.6**

histogram(~nrgy|year,df,xlab = "year")

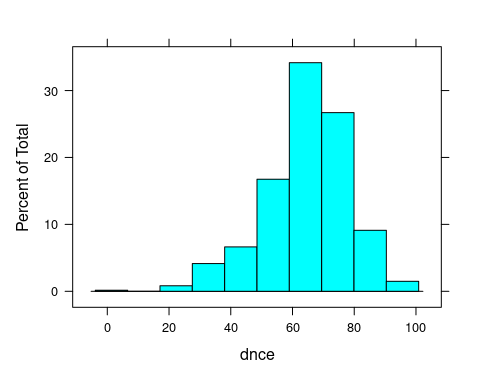


**Fig.1.7**

unique(df["year"])

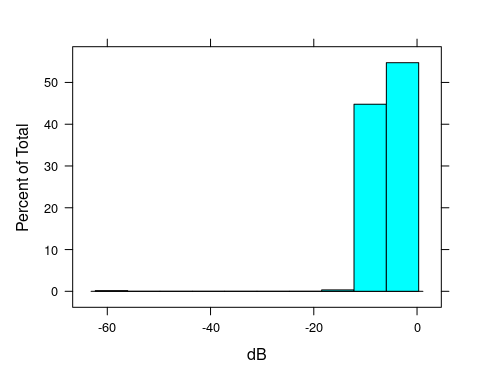
## # A tibble: 10 × 1  
## year  
## <dbl>  
## 1 2010  
## 2 2011  
## 3 2012  
## 4 2013  
## 5 2014  
## 6 2015  
## 7 2016  
## 8 2017  
## 9 2018  
## 10 2019

histogram(~dnce,data=top10s)



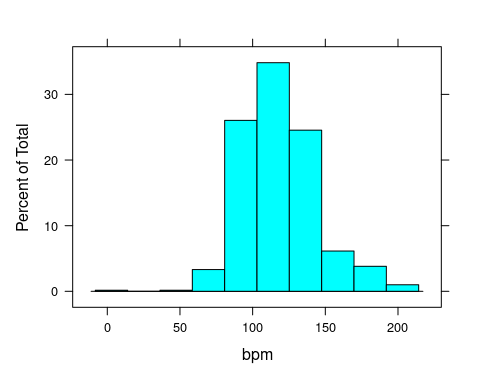
**Fig.1.8**

histogram(~dB,data=top10s)

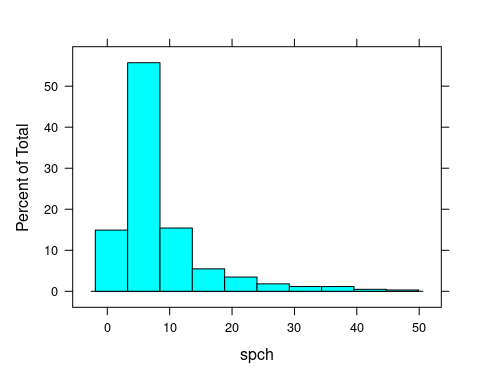


**Fig.1.9**

histogram(~bpm,data=top10s)

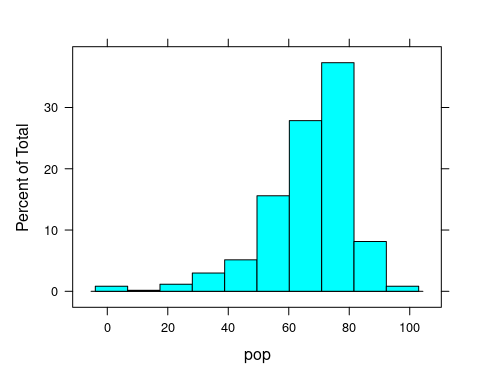
  
 **Fig.1.10**

histogram(~spch,data=top10s)



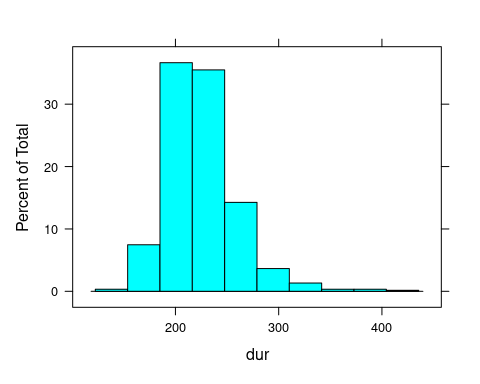
**Fig.1.11**

histogram(~pop,data=top10s)



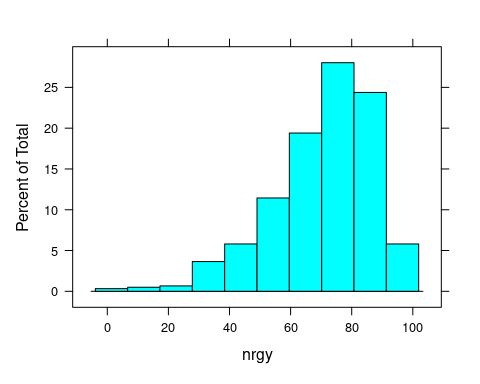
**Fig.1.12**

histogram(~dur,data=top10s)



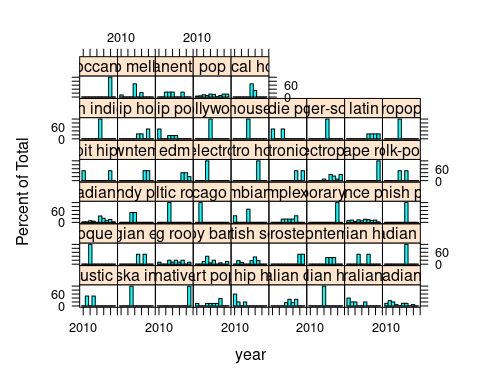
**Fig.1.13**

histogram(~nrgy,data=top10s)



**Fig.1.14**

histogram(~year|top.genre,data=top10s)



**Fig.1.15**

library(plyr)

## ------------------------------------------------------------------------------  
## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)  
## ------------------------------------------------------------------------------  
##   
## Attaching package: 'plyr'  
##   
## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

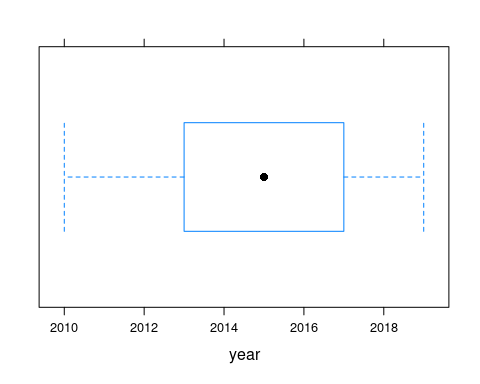
count(top10s,'artist')

## artist freq  
## 1 3OH!3 1  
## 2 5 Seconds of Summer 1  
## 3 A Great Big World 1  
## 4 Adam Lambert 2  
## 5 Adele 10  
## 6 Alan Walker 1  
## 7 Alessia Cara 4  
## 8 Alesso 1  
## 9 Alicia Keys 6  
## 10 Ansel Elgort 1  
## 11 Ariana Grande 9  
## 12 Austin Mahone 2  
## 13 Avicii 3  
## 14 Avril Lavigne 2  
## 15 Bastille 1  
## 16 Bebe Rexha 2  
## 17 Beyonc\xe9 8  
## 18 Birdy 5  
## 19 Britney Spears 9  
## 20 Bruno Mars 13  
## 21 B\xd8RNS 1  
## 22 Calvin Harris 10  
## 23 Camila Cabello 3  
## 24 Cardi B 3  
## 25 Carly Rae Jepsen 5  
## 26 Cashmere Cat 2  
## 27 Charli XCX 1  
## 28 Charlie Puth 3  
## 29 Chris Brown 3  
## 30 Christina Aguilera 6  
## 31 Christina Perri 3  
## 32 Ciara 1  
## 33 Clean Bandit 2  
## 34 CNCO 1  
## 35 Coldplay 4  
## 36 Daddy Yankee 1  
## 37 Daft Punk 2  
## 38 Dan + Shay 1  
## 39 David Guetta 9  
## 40 Demi Lovato 8  
## 41 Disclosure 1  
## 42 DJ Khaled 3  
## 43 DJ Snake 2  
## 44 DNCE 7  
## 45 Drake 2  
## 46 Dua Lipa 3  
## 47 Ed Sheeran 11  
## 48 Ellie Goulding 5  
## 49 Emeli Sand\xe9 3  
## 50 Eminem 2  
## 51 Enrique Iglesias 5  
## 52 Far East Movement 1  
## 53 Fergie 3  
## 54 Fifth Harmony 4  
## 55 Flo Rida 2  
## 56 Florence + The Machine 3  
## 57 fun. 2  
## 58 G-Eazy 2  
## 59 Galantis 1  
## 60 Gwen Stefani 3  
## 61 Gym Class Heroes 1  
## 62 Hailee Steinfeld 3  
## 63 Halsey 1  
## 64 Harry Styles 1  
## 65 Hayley Kiyoko 1  
## 66 Hilary Duff 2  
## 67 Hot Chelle Rae 1  
## 68 Hozier 1  
## 69 Icona Pop 1  
## 70 Iggy Azalea 1  
## 71 J Balvin 1  
## 72 James Arthur 1  
## 73 Janet Jackson 1  
## 74 Jason Derulo 4  
## 75 Jennifer Hudson 1  
## 76 Jennifer Lopez 10  
## 77 Jess Glynne 1  
## 78 Jessie J 3  
## 79 Jewel 1  
## 80 Joey Montana 1  
## 81 John Legend 2  
## 82 John Newman 1  
## 83 Jonas Blue 1  
## 84 Jonas Brothers 3  
## 85 Justin Bieber 16  
## 86 Justin Timberlake 9  
## 87 Kanye West 1  
## 88 Katy Perry 17  
## 89 Kelly Clarkson 5  
## 90 Kelly Rowland 1  
## 91 Kesha 9  
## 92 Khalid 1  
## 93 Kygo 4  
## 94 Labrinth 1  
## 95 Lady Gaga 14  
## 96 Lana Del Rey 4  
## 97 Lea Michele 1  
## 98 Lewis Capaldi 1  
## 99 Liam Payne 3  
## 100 Lilly Wood and The Prick 1  
## 101 Lily Allen 1  
## 102 Little Mix 5  
## 103 Lizzo 2  
## 104 LMFAO 2  
## 105 Lorde 1  
## 106 Lost Frequencies 2  
## 107 Luis Fonsi 1  
## 108 Lukas Graham 1  
## 109 Mabel 1  
## 110 Macklemore & Ryan Lewis 3  
## 111 Madonna 2  
## 112 MAGIC! 1  
## 113 Major Lazer 2  
## 114 Mariah Carey 2  
## 115 Mark Ronson 3  
## 116 Maroon 5 15  
## 117 Marshmello 2  
## 118 Martin Garrix 2  
## 119 Martin Solveig 1  
## 120 Meghan Trainor 6  
## 121 Michael Jackson 1  
## 122 Mike Posner 3  
## 123 Miley Cyrus 5  
## 124 Missy Elliott 2  
## 125 Mr. Probz 1  
## 126 M\xd8 1  
## 127 N.E.R.D 1  
## 128 Naughty Boy 2  
## 129 Ne-Yo 1  
## 130 Nelly Furtado 1  
## 131 Neon Trees 2  
## 132 Niall Horan 2  
## 133 Nick Jonas 4  
## 134 Nicki Minaj 6  
## 135 Olly Murs 1  
## 136 One Direction 7  
## 137 OneRepublic 9  
## 138 Owl City 1  
## 139 P!nk 6  
## 140 Paloma Faith 1  
## 141 Passenger 2  
## 142 Pharrell Williams 2  
## 143 Pitbull 11  
## 144 R3HAB 1  
## 145 RedOne 1  
## 146 Ricky Martin 1  
## 147 Rihanna 15  
## 148 Rita Ora 2  
## 149 Robin Schulz 2  
## 150 Robin Thicke 2  
## 151 Rudimental 2  
## 152 Sam Smith 3  
## 153 Sara Bareilles 1  
## 154 Sean Kingston 1  
## 155 Selena Gomez 8  
## 156 Selena Gomez & The Scene 2  
## 157 SHAED 1  
## 158 Shakira 3  
## 159 Shawn Mendes 11  
## 160 Sia 6  
## 161 Sigala 2  
## 162 Silk City 1  
## 163 Sleeping At Last 1  
## 164 Snakehips 1  
## 165 Swedish House Mafia 1  
## 166 T.I. 2  
## 167 Taio Cruz 2  
## 168 Taylor Swift 8  
## 169 The Black Eyed Peas 5  
## 170 The Chainsmokers 11  
## 171 The Script 1  
## 172 The Wanted 3  
## 173 The Weeknd 5  
## 174 Tinie Tempah 2  
## 175 Tove Lo 2  
## 176 Train 1  
## 177 Troye Sivan 1  
## 178 Usher 2  
## 179 will.i.am 1  
## 180 Wiz Khalifa 1  
## 181 Years & Years 1  
## 182 Zara Larsson 1  
## 183 ZAYN 5  
## 184 Zedd 6

count(top10s,'top.genre')

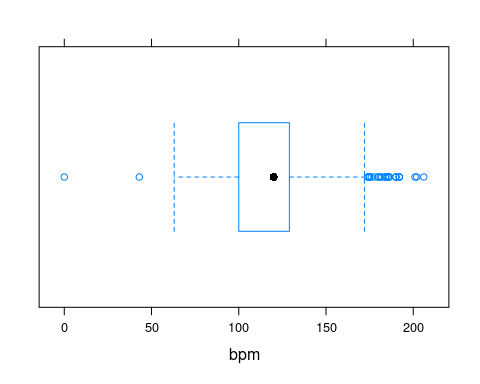
## top.genre freq  
## 1 acoustic pop 2  
## 2 alaska indie 1  
## 3 alternative r&b 1  
## 4 art pop 8  
## 5 atl hip hop 5  
## 6 australian dance 6  
## 7 australian hip hop 1  
## 8 australian pop 5  
## 9 barbadian pop 15  
## 10 baroque pop 2  
## 11 belgian edm 2  
## 12 big room 10  
## 13 boy band 15  
## 14 british soul 11  
## 15 brostep 2  
## 16 canadian contemporary r&b 9  
## 17 canadian hip hop 2  
## 18 canadian latin 1  
## 19 canadian pop 34  
## 20 candy pop 2  
## 21 celtic rock 1  
## 22 chicago rap 1  
## 23 colombian pop 3  
## 24 complextro 6  
## 25 contemporary country 1  
## 26 dance pop 327  
## 27 danish pop 1  
## 28 detroit hip hop 2  
## 29 downtempo 2  
## 30 edm 5  
## 31 electro 2  
## 32 electro house 1  
## 33 electronic trap 2  
## 34 electropop 13  
## 35 escape room 2  
## 36 folk-pop 2  
## 37 french indie pop 1  
## 38 hip hop 4  
## 39 hip pop 6  
## 40 hollywood 1  
## 41 house 1  
## 42 indie pop 2  
## 43 irish singer-songwriter 1  
## 44 latin 4  
## 45 metropopolis 1  
## 46 moroccan pop 1  
## 47 neo mellow 9  
## 48 permanent wave 4  
## 49 pop 60  
## 50 tropical house 3

#Box plot  
bwplot(~year,data=top10s)



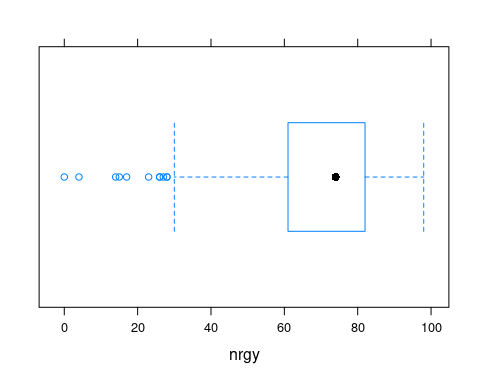
**Fig.2.1**

bwplot(~bpm,data=top10s)



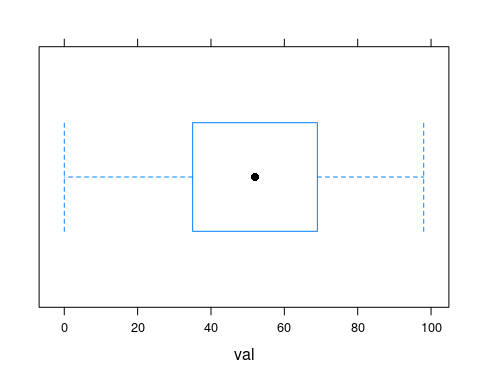
**Fig.2.2**

bwplot(~nrgy,data=top10s)



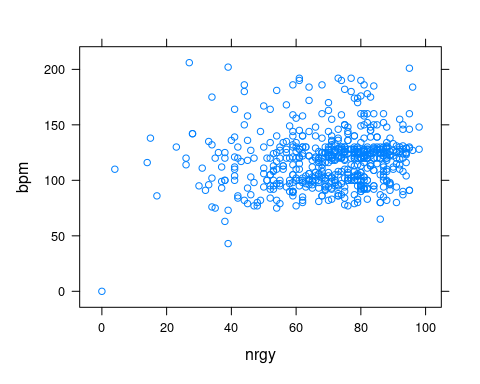
**Fig.2.3**

bwplot(~val,data=top10s)



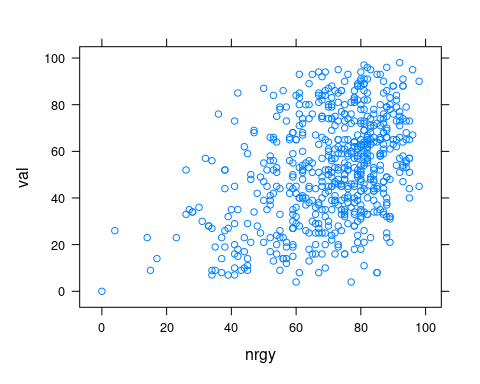
**Fig.2.4**

#Scatter plot  
xyplot(bpm~nrgy,data=top10s)



**Fig.3.1**

xyplot(val~nrgy,data=top10s)



**Fig.3.2**

**Insight and Inference:**

Fig.1: There are more number of dance pop songs is released in 2014. In 2012 and 2018 less number of dance pop song released. The remaining year has medium number of dance pop song.

Fig1.2: The beats per minute(bpm) of the dance pop is mostly lies between 120 to 130.

Fig1.3: The histogram is negative skew. The high energy level of dance pop lies between 70 to 80.

Fig1.4: The valence of the dance pop mostly lies between 60 to 80.

Fig.1.6: Most of the songs released in 2014 to 2016

Fig1.8: High dance level lies between 60 to 80. And the histogram is negative skew.

Fig1.10:Most of the song bpm lies between 100 to 150

Fig1.11: The histogram is positive skew. And most of the song’s speechiness lies between 0 to 10.

Fig1.12: The histogram is negative skew. The high value of pop of the song lies between 60 to 80

Fig2.1: The interquartile lies between 2013 to 2017.

Fig2.2: The interquartile lies between 90 to 120. There is low and high outliers are in beats per minute.

Fig2.3: The interquartile lies between 60 to 80. There is low outliers in energy.

Fig2.4: The interquartile lies between 30 to70. There is no outliers in valence.

Fig3.1,3.2: There is no correlation between val and energy.